

## PATENT APPLICATION TRANSMITTAL LETTER

1c6037 U.S. PTO

(Large Entity)

Docket No.  
INTL-0364-US (P8583)

05/31/00

TO THE ASSISTANT COMMISSIONER FOR PATENTS

Transmitted herewith for filing under 35 U.S.C. 111 and 37 C.F.R. 1.53 is the patent application of:

SCOTT A. ROSENBERG

For: TRANSFORMING PIXEL DATA AND ADDRESSES



Enclosed are:

- ☒ Certificate of Mailing with Express Mail Mailing Label No. EL542038155US
- ☒ Six (6) sheets of drawings.
- ☐ A certified copy of a application.
- ☒ Declaration ☒ Signed. ☐ Unsigned.
- ☒ Power of Attorney
- ☐ Information Disclosure Statement
- ☐ Preliminary Amendment
- ☒ Other: Recordation Form Cover Sheet; Assignment and check for \$40.

## CLAIMS AS FILED

For	#Filed	#Allowed	#Extra	Rate	Fee
Total Claims	25	- 20 =	5	x \$18.00	\$90.00
Indep. Claims	3	- 3 =	0	x \$78.00	\$0.00
Multiple Dependent Claims (check if applicable) <input type="checkbox"/>					\$0.00
BASIC FEE					\$690.00
TOTAL FILING FEE					\$780.00

- ☒ A check in the amount of \$780.00 to cover the filing fee is enclosed.
- ☒ The Commissioner is hereby authorized to charge and credit Deposit Account No. 20-1504 as described below. A duplicate copy of this sheet is enclosed.
- ☐ Charge the amount of as filing fee.
- ☒ Credit any overpayment.
- ☒ Charge any additional filing fees required under 37 C.F.R. 1.16 and 1.17.
- ☐ Charge the issue fee set in 37 C.F.R. 1.18 at the mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.311(b).

Dated: May 31, 2000

Signature  
Timothy N. Trop, Reg. No. 28,994  
TROP, PRUNER & HU, P.C.  
8554 Katy Freeway, Suite 100  
Houston, Texas 77024  
Phone: (713) 468-8880  
Fax: (713) 468-8883

CC:

Customer No. 21906

**APPLICATION**

**FOR**

**UNITED STATES LETTERS PATENT**

**TITLE:           TRANSFORMING PIXEL DATA AND ADDRESSES**

**INVENTOR:       SCOTT A. ROSENBERG**

Express Mail No.: EL542038155US

Date: May 31, 2000

TRANSFORMING PIXEL DATA AND ADDRESSES

Background

This invention relates generally to graphics or video engines commonly used in computer systems.

5 A graphics or video engine is responsible for processing or manipulating data received from a pixel source such as a processor, a graphics controller or other devices. The graphics/video engine takes the pixel data and operates on that pixel data in a variety of well known ways. For example, the color space of the pixel data may  
10 be converted using routine algorithms. Alternatively, the pixel data may be scaled to change the size of the displayed information, again using well known algorithms. Similarly, the pixel data may be subjected to composition effects such as blurring, contrasting or other distortions.

15 In the conventional process, pixels are generated and deposited into memory as a result of a drawing operation such as three dimensional or video based rendering. To impose additional transformations like color space conversion or scaling, these pixels are then typically  
20 fetched from memory by a "fetch" engine. The operation is imposed and the pixels are then written back to the same memory location. The imposition of the transformation is

termed "active" because it requires an explicit fetch engine to be set up with the parameters of the operation.

Thus, when a number of transformations are involved with given pixel data, many fetch engines may be needed that have the redundant functionality of generating pixel addresses. The use of these fetch engines complicates the memory controller that must operate with all of the fetch engines contending for memory bandwidth.

The need to use the fetch engines may also reduce the available memory bandwidth because of the need for read-modify-write cycles when imposing filtering operations after rendering. Moreover, when multiple transformations are needed, the programmer must awkwardly impose the transformations by causing the fetch engine to manipulate the data between a memory location and the various transformation engines. In addition, there is no easy way to cause multiple transformations to occur in a serial fashion.

Thus, there is a need for better ways to impose transformations on pixel data in graphic/video engines.

#### Brief Description of the Drawings

Figure 1 is a schematic depiction of the prior art;

Figure 2 is a schematic depiction of one embodiment of the present invention;

Figure 3 is a more detailed schematic depiction of one embodiment of the present invention;

Figure 4 shows the memory architecture of one embodiment of the present invention;

Figure 5 is a flow chart for software useful in accordance with one embodiment of the present invention;

5        Figure 6 is a flow chart useful with another embodiment of the present invention;

Figure 7 shows a memory architecture in accordance with an embodiment of the present invention; and

10        Figure 8 is a block depiction of a processor-based system in accordance with one embodiment of the present invention.

#### Detailed Description

Referring to Figure 1, in the conventional active system of imposing transformations on pixel data, the pixel data may be operated on by a plurality of transformation engines such as the scaling engine 100, the color conversion engine 106, and the composition engine 104. To do so, a fetch engine (not shown) fetches the data from memory and passes it to a first transformation engine, such as the scaling engine 100, as indicated by the arrow marked 1. When the transformation is complete, the fetch engine returns the data to the memory 102 as indicated by the arrow 2. Similarly, the data may be shuttled between the color conversion engine 106 and the composition engine 104 by using a fetch engine to fetch the data from memory 102,

15  
20  
25

and then fetch the data back from memory 102, normally to the same memory 102 location.

In accordance with one embodiment of the present invention, shown in Figure 2, a serial architecture for pixel data transformations may be implemented. For example, the data may be fetched from memory 102 into a scaling engine 100 as indicated by the arrow 1. When the scaling is complete, the addresses for the data may be transformed to write the data to a color conversion engine 106 as indicated by the arrow 2. After color conversion, the data addresses may be transformed so that the data is written directly into a composition engine 104. Thereafter, the composed data may be written back to the memory 102 as indicated by the arrow 4. Thus, the operations accomplished with embodiments of the present invention may be more efficient than those of the prior art illustrated in Figure 1.

The pixel data operations in accordance with one embodiment of the present invention may be termed passive as compared to the active operations utilized in the prior art. An application may write pixels to a range of virtual memory addresses and the passive engine may impose the chosen operation. A new "re-mapped" memory address is generated and the pixel data is written to the new memory location.

Any conventional transformation or operation performed on pixel data including but not limited to, scaling, color conversion and composition can be implemented using a passive engine. Moreover, as indicated in Figure 2, the multiple transformations may be tied together and performed in a serial fashion by setting the output addresses of one function to the same range as the input address of another function.

As shown in Figure 3, a pixel source 12, that may be a processor, a graphics controller or another device, sends data and address information to a memory controller 14. Some of the information passed from the pixel source 12 to the memory controller 14 may map to a virtual memory range of a media port target 16. The media port target 16 is a memory controller client that accepts pixel data written to virtual addresses set aside by the operating system.

The media port target 16 may provide the data to a variety of transfer functions 18 whose purpose is to receive pixel data and addresses from the media port target 16, perform a pixel and address transformation, and forward output pixel data and addresses to a media port write back engine 20. The media port write back engine 20 is a write-only memory controller client that receives pixel and address information from the individual transfer functions 18 and forwards the pixel data and addresses to the memory controller 14.

The operating system cooperates with the media port target 16 to set up a range of virtual memory addresses for use by the media port target 16. The operating system may set aside a specific memory range as virtual addresses that map to the media port target 16. Pixel data written to this memory range is not forwarded to physical memory. Instead, that pixel data is forwarded to the media port target 16. Each of the transfer functions 18 is also assigned a virtual memory range within the overall virtual memory range set aside for the media port target 16.

Thus, referring to Figure 4, the total addressable memory 24 may include a portion 26 of virtual memory set aside for the media port target 16. Within the memory range 26, dedicated to the media port target 16, are transfer function address ranges  $28_1$  through  $28_n$ , each dedicated one of  $n$  transfer functions 18. Each transfer function 18 defined in the media port target 16 has a defined output memory address range 28. Pixel data written to a transfer function memory address range 28 goes through a transformation and has its addresses translated or re-addressed as indicated at 20.

As a specific hypothetical example, the operating system may set aside the memory address range from 0x000000 to 0x800000 (8 megabytes) as virtual addresses that map to the media port target 16. The range of addresses from 0x000000 to 0x200000 may be set aside for a first transfer



function 18 such as color conversion. The range of  
addresses from 0x200000 to 0x800000 may be set aside for a  
second transfer function 18. The color conversion  
function's output memory address range may be defined to be  
5 from 0x800000 through 0xA00000.

Each transfer function 18 performs a specific pixel  
transformation operation and address translation. The  
pixel data and addresses generated by the various transfer  
functions 18 may be written back to the memory controller  
10 14. While a single memory port write back engine 20 is  
illustrated in Figure 3, more than one media port write  
back engine 20 may be utilized to consolidate the pixel  
data and interfaces to the memory controller 14.

As another specific hypothetical example, the transfer  
15 function 18a may be the color space transformation from the  
RGB color space to YUV color space. For example, the base  
address "baseinput" of the input pixel data may be  
0x000000. The base address "baseoutput" of the transfer  
function 18 output pixel data may be 0x800000. Applying  
20 the transfer function 18, the RGB to YUV conversion  
operation may be as follows:

$$y=a1xR+a2xG+a3xB;$$

$$u=b1xR+b2xG+b3xB;$$

$$v=c1xr+c2xG+c3xB.$$

25 The addresses of the pixel data after color space  
conversion are transformed from their original pixel

addresses. First, the offset to each pixel is determined by subtracting the baseinput from each pixel's original address. Then, the offset is added to the baseoutput to get the new address.

5           In this way, each RGB pixel data may be converted through the transformation equation indicated above to output YUV pixel data. The output YUV pixel data is deposited at memory location 0x800000 to 0xA00000 in an example in which the baseoutput is 0x800000 and the  
10          available memory range stops at 0xA00000. Similarly, the new addresses of the color space transformed pixel data is the offset added to the baseoutput to position the pixel data in easily determined locations, for the next transfer function or for return to memory.

15          As another hypothetical example, a first transfer function, such as a color space conversion, may be implemented in the virtual memory address range from 0x000000 to 0x200000 as indicated in Figure 5. A second transfer function, performing a 2:1 horizontal scaling, may  
20          be implemented in the virtual memory address range from 0x200000 to 0x400000. A third transfer function, performing a blurring operation, may be implemented in the virtual memory address range from 0x400000 to 0x500000.

25          The first through third transfer functions may be performed serially by setting the output memory address range of the first transfer function to start at 0x200000,

and setting the output memory address range of the second transfer function to start at 0x400000. Pixel data written to the range from 0x000000 to 0x200000 is color converted and forwarded to the virtual memory address range from 0x200000 through 0x400000. This pixel data is then scaled by horizontal averaging and written to the virtual memory address range from 0x400000 through 0x500000. The final output data of this sequence may written to the output memory address range designated by the third transfer function.

Transfer function software 32, shown in Figure 6, may be stored in association with the memory controller 14, in accordance with one embodiment of present invention. Initially, the software 32 determines whether the pixel data has been received as indicated in diamond 34. If so, the first transfer function's transformation is performed as indicated in block 36. Thereafter, the address transformation is performed as indicated in block 38.

An example of serialized transformations is shown in Figure 7. The software 40, in accordance with one embodiment of the present invention, determines in diamond 42 when the pixel data has been received. Upon receipt of the pixel data, the first transfer function is performed as indicated in block 44. Thereafter, the output memory address range is set to the base address of the second transfer function as indicated in block 46. The second

transfer function may be performed followed by address transformation as indicated in block 48. The serial execution of transfer functions followed by address transformations may be extended to any number of transfer functions.

Referring to Figure 8, in accordance with one embodiment of the present invention, the memory controller 14 is coupled to system memory 50 and a bridge 56. The bridge 56 is in turn coupled to a processor 58 and a graphics controller 52. The graphics controller 52 may be coupled to a display 54. Thus, the pixel source 12 may be the processor 58, the graphics controller 52 or some other device. In one embodiment of the present invention, the software 32 and 40 may be stored in a storage associated with the memory controller 14. As is conventional, the bridge 56 is coupled to a bus 24. The bus 24 may include a peripheral device 26 as illustrated.

In some embodiments of the present invention, there may be no need to use a plurality of fetch engines having redundant pixel address generating functionality. These fetch engines may complicate the memory controller that arbitrates among all of the engines contending for memory bandwidth. Transfer functions may be implemented, in accordance with embodiments of the present invention, under the same master memory controller although other implementations may also be possible. In addition, memory

bandwidth utilization may be reduced by the elimination of the read-modify-write cycle typically used in active architectures when imposing filtering operations after rendering. In accordance with some embodiments of the present invention, pixel data being generated may be written directly to one or more successive memory-mapped media ports that perform the desired filter operations before returning the pixel data to memory.

In some embodiments of the present invention, multiple filter operations may be flexibly allocated and serially applied through intelligent memory aliasing. If a programmer wants to impose color space transformations, followed by scaling, followed by convolution operations, the programmer may do so by simply mapping one operation's output into the input of the next operation. With conventional active systems, filter operations are typically implemented in the middle of a fetch and write back circuit. If several operations are implemented and only one is used, the other operations are effectively unusable for the duration of the operation because of the use of a single, standalone execution pipe.

While the present invention has been described with respect to a limited number of embodiments, those skilled in the art will appreciate numerous modifications and variations therefrom. It is intended that the appended

claims cover all such modifications and variations as fall within the true spirit and scope of this present invention.

What is claimed is:

1           1.    A method comprising:  
2                writing pixel data to a first memory location;  
3                performing a first pixel transformation at said  
4 first memory location;  
5                generating a memory address for a second memory  
6 location; and  
7                writing said transformed pixel data from said  
8 first memory location to said second memory location.

1           2.    The method of claim 1 wherein writing pixel data  
2 to a first memory location includes writing pixel data to a  
3 first virtual memory location.

1           3.    The method of claim 2 further including writing  
2 pixel data to a virtual memory location associated with a  
3 memory controller client that receives pixel data written  
4 to certain virtual addresses.

1           4.    The method of claim 3 including causing an  
2 operating system to set aside virtual addresses for said  
3 memory controller client.

1           5.    The method of claim 1 wherein entering an address  
2 for a second memory location includes transforming the  
3 addresses of said pixel data at said first memory location  
4 to addresses at said second memory location.

1           6.    The method of claim 5 including determining the  
2   offset to each pixel data by subtracting a base address at  
3   said first memory location from the address of each pixel  
4   data.

1           7.    The method of claim 6 including adding said  
2   offset to a base address of said second memory location.

1           8.    The method of claim 1 wherein writing said  
2   transformed pixel data from said first memory location to  
3   said second memory location includes transferring said  
4   pixel data to a memory controller using a memory controller  
5   client.

1           9.    The method of claim 1 wherein writing said  
2   transformed pixel data from said first memory location to  
3   said second memory location includes writing said pixel  
4   data from a first memory location associated with a first  
5   transfer function to a second memory location associated  
6   with a second transfer function.

1           10.   The method of claim 9 including transforming the  
2   addresses of said pixel data from addresses in a first  
3   virtual memory range associated with said first transfer



4 function to memory addresses in a second virtual memory  
5 range associated with said second transfer function.

1 11. An article comprising a medium storing  
2 instructions that enable a processor-based system to:  
3 write pixel data to a first memory location;  
4 perform a first pixel transformation at said  
5 first memory location;  
6 generate a memory address for a second memory  
7 location; and  
8 write said transformed pixel data from said first  
9 memory location to said second memory location.

1 12. The article of claim 11 further storing  
2 instructions that enable the processor-based system to  
3 write pixel data to a first virtual memory location.

1 13. The article of claim 12 further storing  
2 instructions that enable the processor-based system to  
3 write pixel data to a virtual memory location associated  
4 with a memory controller client that receives pixel data  
5 written to certain virtual addresses.

1 14. The article of claim 13 further storing  
2 instructions that enable the processor-based system to

3 cause an operating system to set aside virtual addresses  
4 for said memory controller client.

1 15. The article of claim 11 further storing  
2 instructions that enable the processor-based system to  
3 transform the addresses of pixel data at said first memory  
4 location to addresses at said second memory location.

1 16. The article of claim 15 further storing  
2 instructions that enable the processor-based system to  
3 determine the offset to each pixel data by subtracting a  
4 base address at said first memory location from the address  
5 of each pixel data.

1 17. The article of claim 16 further storing  
2 instructions that enable the processor-based system to add  
3 said offset to a base address of said second memory  
4 location.

1 18. The article of claim 11 further storing  
2 instructions that enable the processor-based system to  
3 transfer said pixel data to a memory controller using a  
4 memory controller client.

1 19. The article of claim 11 further storing  
2 instructions that enable the processor-based system to

3 write said pixel data from a first memory location  
4 associated with a first transfer function to a second  
5 memory location associated with a second transfer function.

1 20. The article of claim 19 further storing  
2 instructions that enable the processor-based system to  
3 transform the addresses of said pixel data from addresses  
4 in a first virtual memory range associated with said first  
5 transfer function to memory addresses in a second virtual  
6 memory range associated with said second transfer function.

1 21. A system comprising:  
2 a memory controller that receives pixel data and  
3 addresses;  
4 a first memory controller client that forwards  
5 pixel data and addresses to a first transfer function; and  
6 a second memory controller client that receives  
7 data from said first transfer function together with new  
8 addresses.

1 22. The system of claim 21 wherein said first  
2 controller client selectively forwards pixel data and  
3 addresses to one of a plurality of transfer functions and  
4 said second controller client receives pixel data with new  
5 addresses from a plurality of transfer functions.

1           23. The system of claim 22 wherein said memory  
2 controller client writes the pixel data back to said memory  
3 controller.

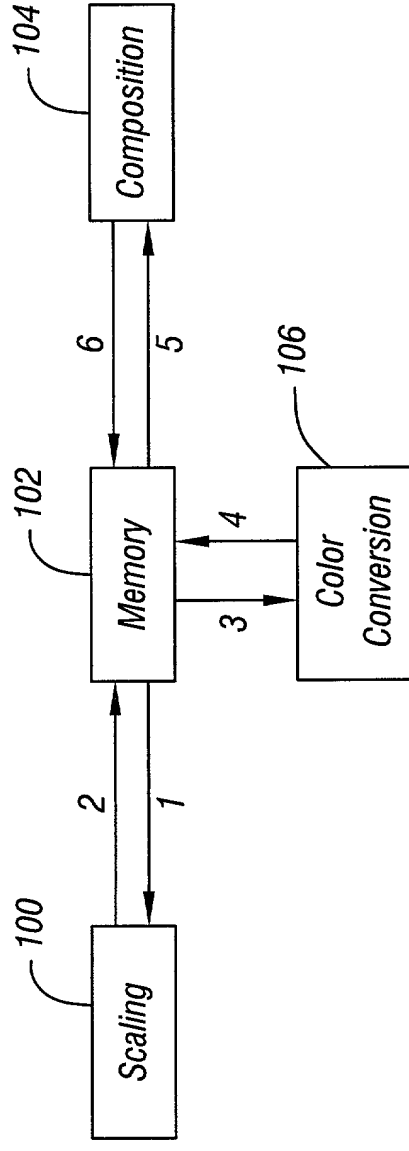
1           24. The system of claim 21 including a plurality of  
2 transfer functions, one of said transfer functions arranged  
3 to write output data to an address range of another  
4 transfer function.

1           25. The system of claim 24 wherein said transfer  
2 functions are associated with virtual memory address  
3 ranges.

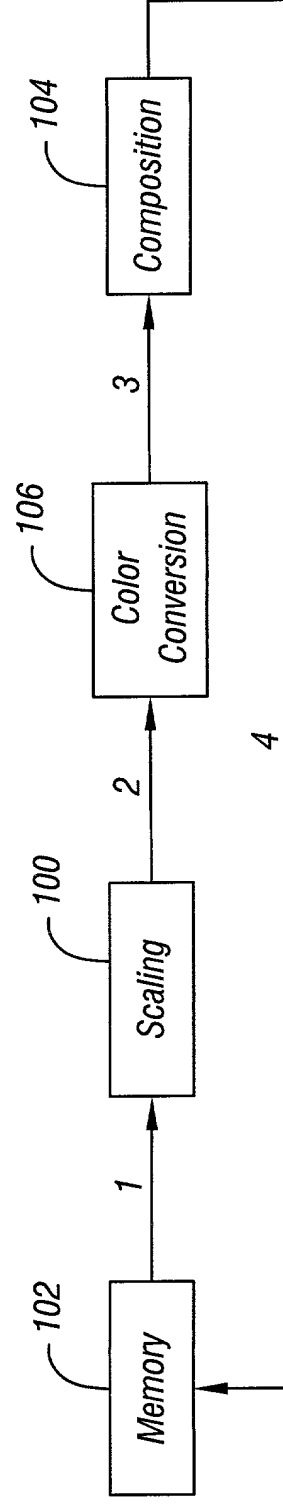
## TRANSFORMING PIXEL DATA AND ADDRESSES

### Abstract of the Disclosure

In a passive pixel data handling system, pixel data may be transferred to a transfer function, at a given address range. The transfer function may perform a transformation and readdress the pixel data. For example, the data may be received through a media port target which transfers the pixel data to a transfer function located at an address range in virtual memory. Each transfer function may readdress the pixel data and forward it to a media port write back engine or to the memory address range of another transfer function.



**FIG. 1**  
**Prior Art**



**FIG. 2**

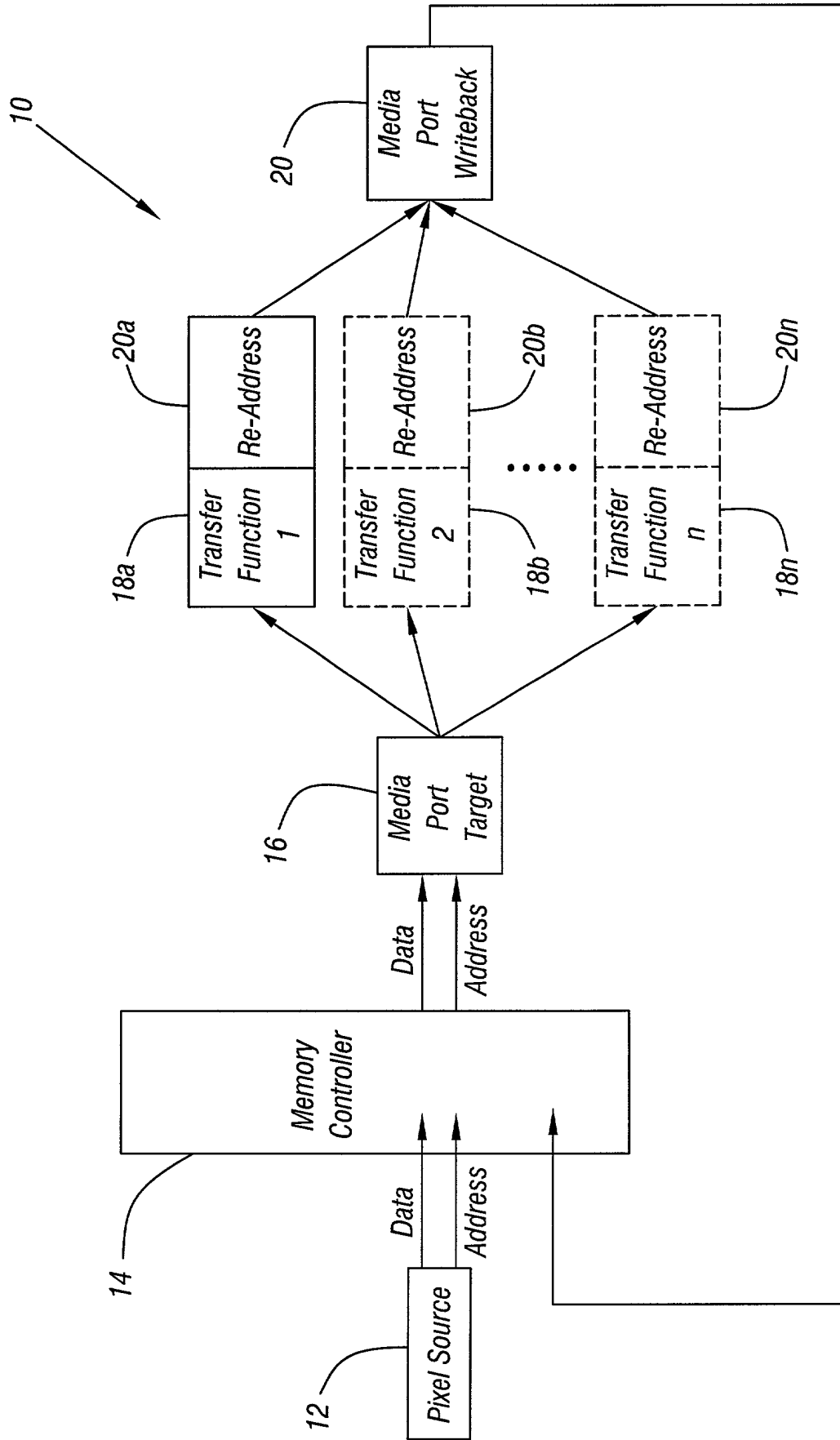
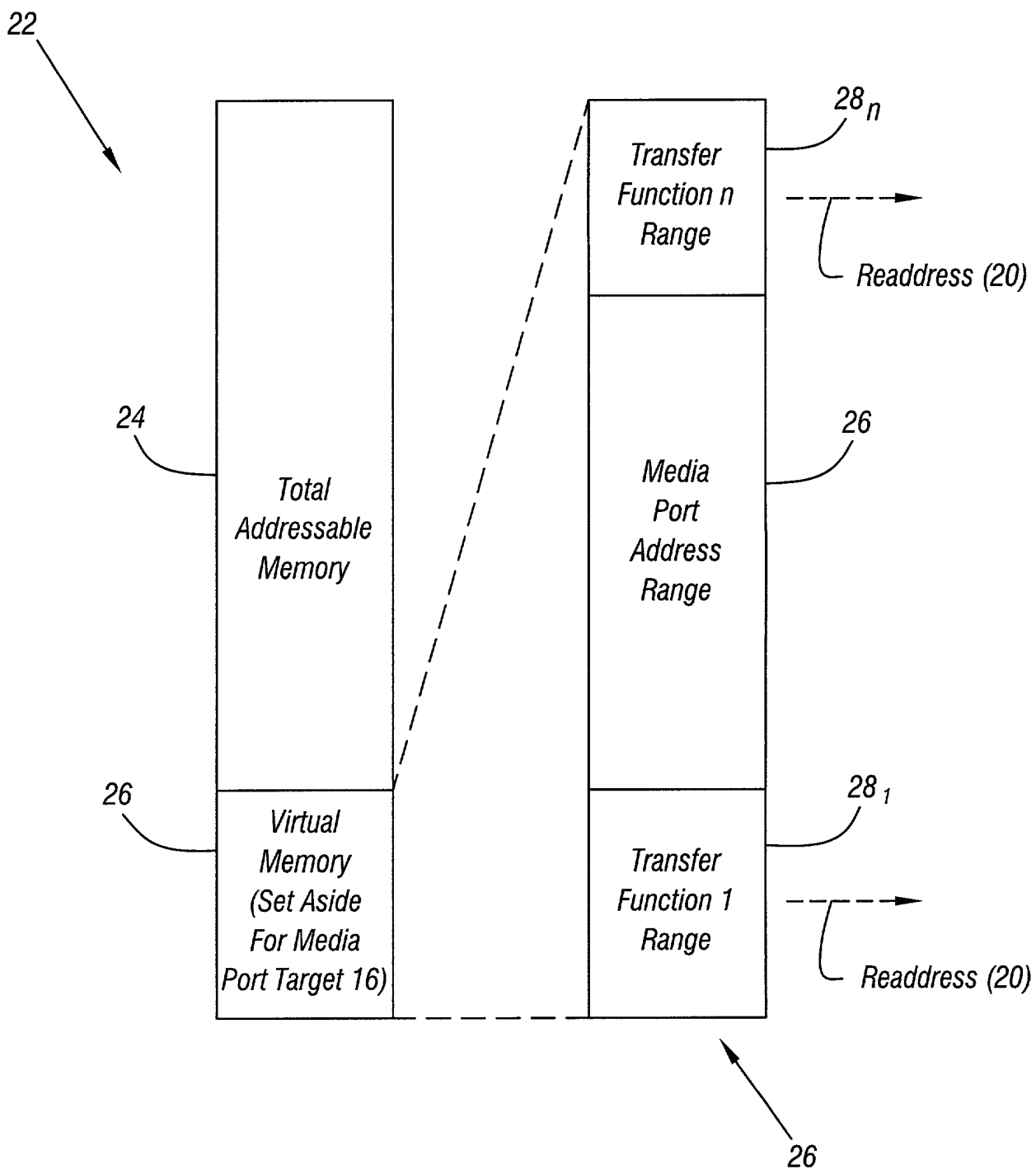
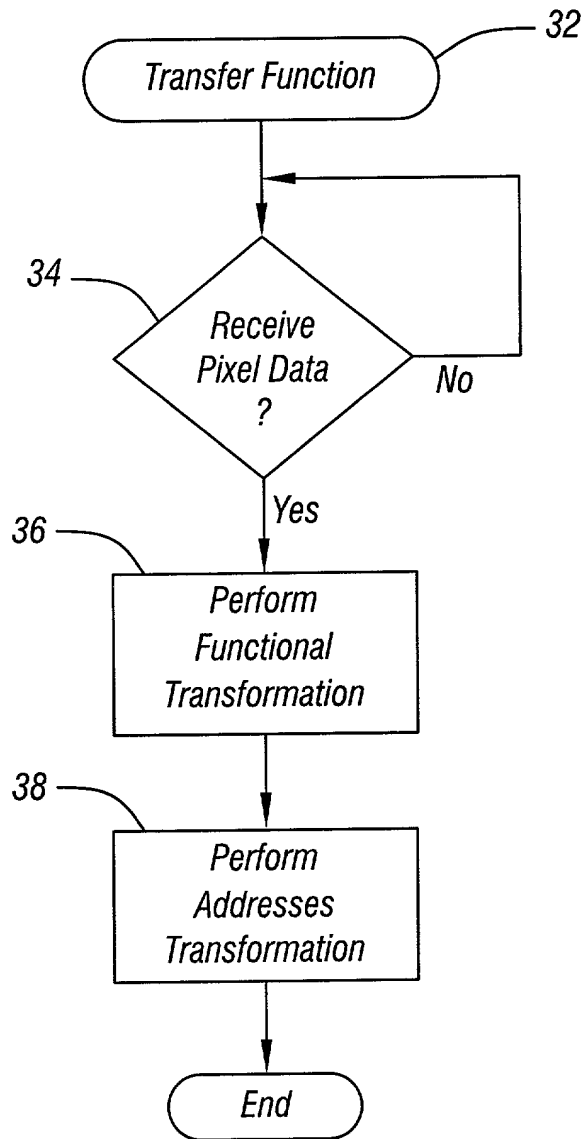


FIG. 3

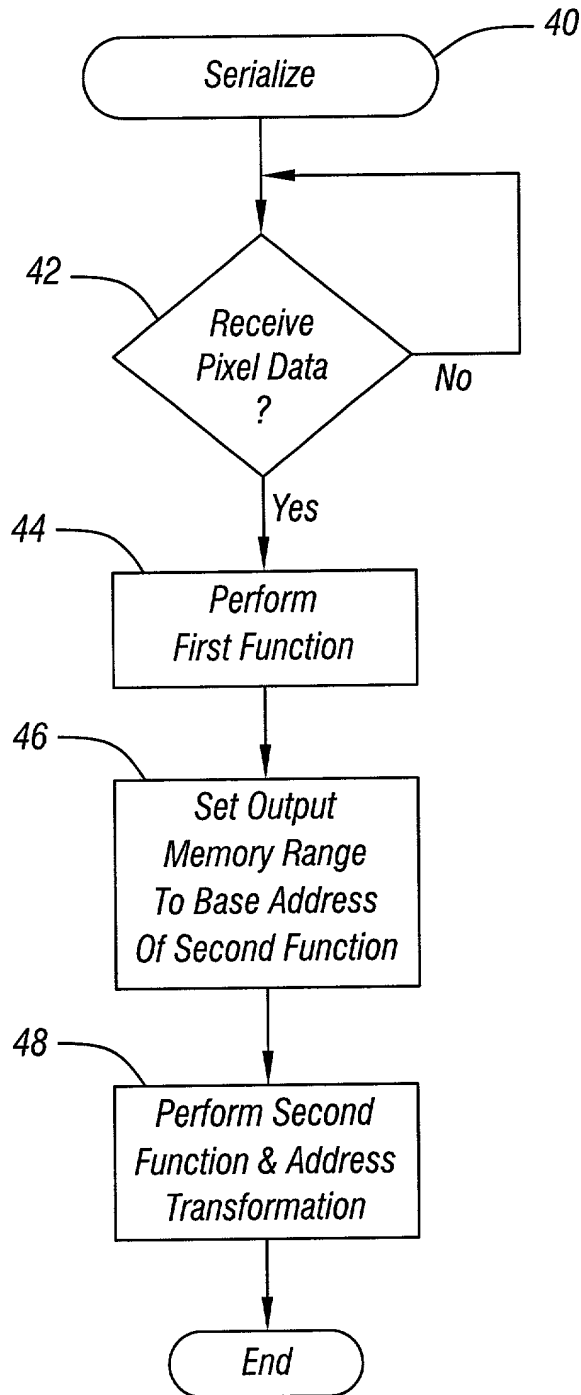


**FIG. 4**

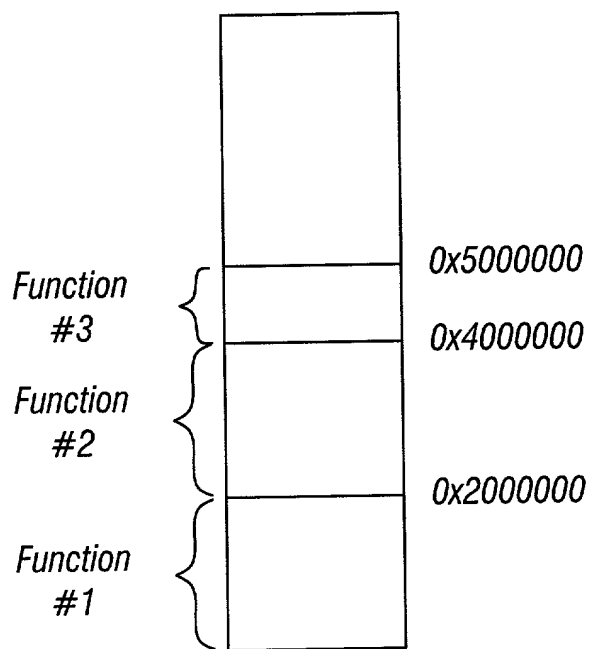




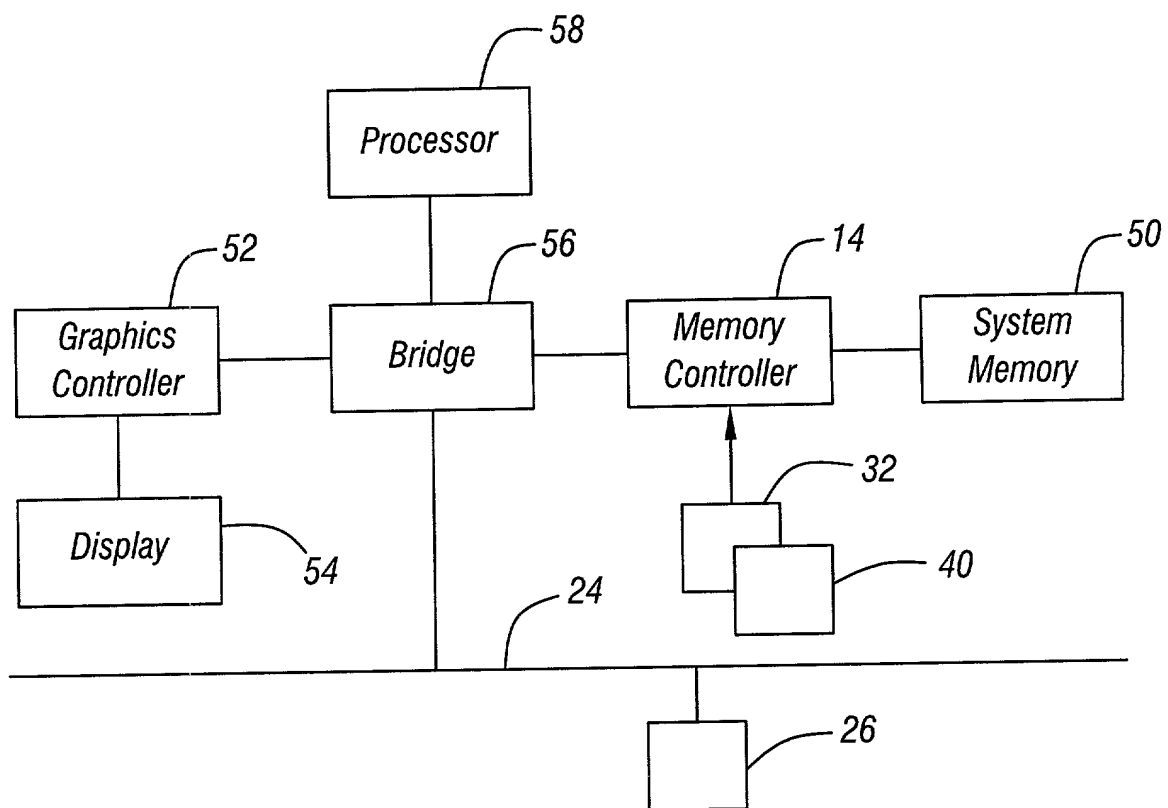
**FIG. 6**



**FIG. 7**



**FIG. 5**



**FIG. 8**

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below, next to my name.

I believe I am the original, first, and sole inventor (if only one name is listed below) or an original, first, and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

TRANSFORMING PIXEL DATA AND ADDRESSES

the specification of which

X	is attached hereto.
	was filed on _____ as
	United States Application Number _____
	or PCT International Application Number _____
	and was amended on _____
	(if applicable)

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claim(s), as amended by any amendment referred to above. I do not know and do not believe that the claimed invention was ever known or used in the United States of America before my invention thereof, or patented or described in any printed publication in any country before my invention thereof or more than one year prior to this application, that the same was not in public use or on sale in the United States of America more than one year prior to this application, and that the invention has not been patented or made the subject of an inventor's certificate issued before the date of this application in any country foreign to the United States of America on an application filed by me or my legal representatives or assigns more than twelve months (for a utility patent application) or six months (for a design patent application) prior to this application.

I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119(a)-(d), of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

<u>Prior Foreign Application(s):</u>			<u>Priority Claimed</u>	
Number	(Country)	(Day/Month/Year Filed)	Yes	No
Number	(Country)	(Day/Month/Year Filed)	Yes	No
Number	(Country)	(Day/Month/Year Filed)	Yes	No

I hereby claim the benefit under title 35, United States Code, Section 119(e) of the United States provisional application(s) listed below:

_____ (Application Number)	_____ (Filing Date)
_____ (Application Number)	_____ (Filing Date)

I hereby claim the benefit under Title 35, United States Code, Section 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, Section 112, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal regulations, Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of this application:

_____ (Application Number)	_____ Filing Date	_____ (Status-patented, pending, abandoned)
_____ (Application Number)	_____ Filing Date	_____ (Status-patented, pending, abandoned)

I hereby appoint Timothy N. Trop, Reg. No. 28,994; Fred G. Pruner, Jr., Reg. No. 40,779 and Dan C. Hu, Reg. No. 40,025 my patent attorneys, of TROP, PRUNER & HU, P.C., with offices located at 8554 Katy Freeway, Ste. 100, Houston, TX 77024, telephone (713) 468-8880, and Joseph R. Bond, Reg. No. 36,458; Richard C. Calderwood, Reg. No. 35,468; Sean Fitzgerald, Reg. No. 32,027; David J. Kaplan, Reg. No. 41,105; Leo V. Novakoski, Reg. No. 37,198; Naomi Obinata, Reg. No. 39,320; Thomas C. Reynolds, Reg. No. 32,488; Steven P. Skabrat, Reg. No. 36,279; Howard A. Skaist, Reg. No. 36,008; Steven C. Stewart, Reg. No. 33,555; Raymond J. Werner, Reg. No. 34,752; and Charles K. Young, Reg. No. 39,425; my patent attorneys, of INTEL CORPORATION; with full power of substitution and revocation, to prosecute this application and to transact all business in the Patent and Trademark Office connected herewith.

Send correspondence to Timothy N. Trop, TROP, PRUNER & HU, P.C., 8554 Katy Freeway, Ste. 100, Houston, TX 77024 and direct telephone calls to Timothy N. Trop, (713) 468-8880.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

